Deconfounded Lifelong Learning for Autonomous Driving via Dynamic Knowledge Spaces

Jiayuan Du*, Yuebing Song*, Yiming Zhao, Xianghui Pan, Jiawei Lian, Yuchu Lu, Liuyi Wang, Chengju Liu†, and Qijun Chen†

Tongji University, Shanghai, China
{dujiayuan, yuebing_song, liuchengju, qjchen}@tongji.edu.cn
* Equal contributions † Corresponding author

Abstract. End-to-End autonomous driving (E2E-AD) systems face challenges in lifelong learning, including catastrophic forgetting, difficulty in knowledge transfer across diverse scenarios, and spurious correlations between unobservable confounders and true driving intents. To address these issues, we propose DeLL, a Deconfounded Lifelong Learning framework that integrates a Dirichlet process mixture model (DPMM) with the front-door adjustment mechanism from causal inference. The DPMM is employed to construct two dynamic knowledge spaces: a trajectory knowledge space for clustering explicit driving behaviors and an implicit feature knowledge space for discovering latent driving abilities. Leveraging the non-parametric Bayesian nature of DPMM, our framework enables adaptive expansion and incremental updating of knowledge without predefining the number of clusters, thereby mitigating catastrophic forgetting. Meanwhile, the front-door adjustment mechanism utilizes the DPMM-derived knowledge as mediators to deconfound spurious correlations, such as those induced by sensor noise or environmental changes, and enhances the causal expressiveness of the learned representations. Additionally, we introduce an evolutionary trajectory decoder that enables non-autoregressive planning. To evaluate the lifelong learning performance of E2E-AD, we propose new evaluation protocols and metrics based on Bench2Drive. Extensive evaluations in the closed-loop CARLA simulator demonstrate that our framework significantly improves adaptability to new driving scenarios and overall driving performance, while effectively retaining previously acquired knowledge.

Keywords: End-to-End Autonomous Driving · Lifelong Learning · CARLA

1 Introduction

End-to-End autonomous driving (E2E-AD) methods [15, 17, 18, 40, 46, 57] have achieved remarkable performance in closed-loop CARLA [11] simulator. However, their deployment in open-world, non-stationary environments is severely hindered by catastrophic forgetting and causal confusion [7, 10]. Imitation learning architectures, operating primarily as correlational engines, struggle to continuously assimilate new scenarios without overwriting historical parameters, as

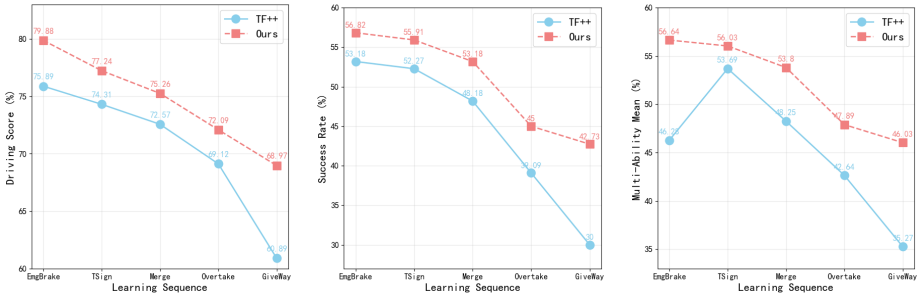

Fig. 1: Driving score, success rate and multi-ability success rate during the lifelong learning process. Our method not only demonstrates superior overall driving performance but also substantially mitigates catastrophic forgetting.

shown in Fig. 1. Furthermore, treating driving as a partially observable Markov decision process reveals that these models frequently capture spurious correlations induced by unobserved confounders, leading to decision-making failures under continuous covariate shifts [42].

Existing methods either improve the modeling ability through transformer-based methods and diverse auxiliary tasks [15,17,21,46,57], or improve the interpretability of the model through interpretable intermediate representations and visualizations [8,19,44], or improve the reasoning ability and trajectory generalization performance of the model by means of large models (LLMs) and diffusion policy [13,30,40,49,52]. However, empowering models with lifelong learning capabilities and mitigating causal confusion issues through dynamic knowledge spaces seems to be unexplored. In the closed-loop simulator CARLA [11], there is also a lack of a lifelong learning benchmark.

To address aforementioned problems, we introduce **DeLL**, a **De**confounded **L**ifelong **L**earning framework for E2E-AD. The proposed architecture resolves the rigidity of fixed-capacity networks by introducing dynamic dual knowledge spaces governed by a Dirichlet process mixture model (DPMM) [28]. As a Bayesian non-parametric model, it dynamically instantiates new cluster components (knowledge anchors) as novel driving scenarios are encountered, preserving historical knowledge in isolated, specialized distributions. Our dynamic knowledge spaces alleviate catastrophic forgetting without relying on rigid task boundaries [35] or computationally expensive experience replay buffers [3]. The dynamically generated knowledge anchors also serve as the mediator variables required for causal front-door adjustment [38, 48] in our causal feature enhancement module. We implement front-door adjustment in an attention-based way to mitigate the spurious correlations of unobservable confounders. By continuously injecting accumulated historical priors back into the network's forward propagation process, it achieves highly efficient knowledge transfer, supporting the lifelong learning goals. We also design the evolutionary trajectory decoder to match our dynamic knowledge spaces. In order to verify the effectiveness of the proposed method, we integrate the evaluation protocols of lifelong learning, design the lifelong

learning task sequence based on Bench2Drive's multi-ability classification, and introduce the corresponding evaluation metrics. Our method outperforms the previous state-of-the-art in both lifelong learning and full-data learning settings.

Our contributions are summarized as follows:

- A novel deconfounded lifelong learning framework "DeLL" for E2E-AD.
- DPMM-based dual knowledge spaces dynamically preserve and update latent feature and trajectory knowledge.
- A causal feature enhancement module leverages knowledge spaces and front-door adjustment to enhance features and alleviates spurious correlations caused by unobservable confounders.
- An evolutionary trajectory decoder that natively supports non-autoregressive, parallel trajectory generation.
- A new lifelong learning evaluation protocol based on the Bench2Drive benchmark, where our method achieves state-of-the-art closed-loop performance.

2 Related Work

2.1 End-to-End Autonomous Driving in CARLA

E2E-AD in CARLA [11] closed-loop simulator is predominantly built upon behavior cloning paradigms. Early works like LBC [6] pioneers the teacher-student distillation approach, transferring privileged information to a vision-only policy. LAV [5] expands this idea by incorporating multi-agent trajectory data to better understand social interactions. TCP [50] proposes trajectory-guided control with uncertainty estimation for safer decision-making.

The advent of transformer models has profoundly reshaped the paradigm of sensor fusion. Transfuser [9] first introduces cross-modal attention mechanisms. Its successor Transfuser++ [17, 57] enhances temporal modeling for occluded scenarios and long horizon planning. InterFuser [44] and E2E-Parking [51] employ a transformer encoder for multi-modal feature fusion and a transformer decoder to generate sequential waypoints. DriveTransformer [21] further unifies perception and planning within transformer frameworks.

To improve the interpretability of end-to-end networks, many works introduce auxiliary tasks. Approaches such as PanT [41], UniAD [15], and VAD [22] emphasize joint optimization of driving sub-tasks. SimLingo [40] further integrates large language models to jointly address closed-loop driving, vision-language understanding, and language-action alignment. ORION [13] integrates neural planners with symbolic guardrails to ensure strict adherence to explicit traffic rules. ThinkTwice [19] employs a two-stage decoder with attention-based rationalization, while HiP-AD [46] uses hierarchical predictive modeling.

Resource-efficient designs have also been explored. AD-MLP [55] replaces transformers with MLP-Mixers for real-time inference. DriveMoE [52] employs dynamic mixture-of-experts routing. DriveAdapter [18] enables plug-and-play adaptation of foundation models. DiffAD [49] leverages diffusion models to address ambiguous scenarios via probabilistic sampling.

However, existing closed-loop E2E-AD methods still face fundamental challenges in incrementally learning new driving abilities in dynamic open environments as humans do. In this paper, we explore how to integrate lifelong learning in an E2E-AD framework.

2.2 Lifelong Learning

Lifelong learning, also known as incremental learning, aims to enable models to acquire new knowledge while retaining previously learned information, with catastrophic forgetting being the core challenge [47]. Existing approaches fall into three categories [34]. Architectural strategies dynamically expand network structures: PNN [43] allocates isolated columns for each task with lateral connections, while ExpertGate [2] trains dedicated experts with a gating network for selection. Regularization strategies constrain parameter updates based on importance: EWC [26] penalizes changes to critical parameters via Fisher information, MAS [1] computes importance from output sensitivity, LwF [29] applies knowledge distillation using old model outputs as supervision, AFA [54] further introduces multi-level feature alignment losses. Rehearsal strategies revisit past data by storing exemplars or generating synthetic samples: iCaRL [3] combines distillation with prototype replay, FearNet [24] mimics biological memory with separate modules for rapid learning and long-term storage, generative models [14, 25] produce pseudo-samples for replay [45, 56].

Most existing lifelong learning methods focus on alleviating forgetting, while paying insufficient attention to the organization of knowledge itself. Moreover, the lifelong learning in the field of closed-loop autonomous driving in CARLA seems to be unexplored.

3 Method

We introduce DeLL, a novel deconfounded lifelong learning framework for E2E-AD. As shown in Fig. 2, the overall workflow of the method features core modules including a multi-modal perception backbone, dynamic dual knowledge spaces, causal feature enhancement modules, and an evolutionary trajectory decoder.

3.1 Multi-modal Perception Backbone

The front end of the network utilizes Transfuser++ [17,57] as the base extractor, leveraging its simplicity and modular extensibility. It first processes RGB image sequences and LiDAR point clouds independently using RegNetY [39] to obtain multi-scale features. Subsequently, cross-modal cues are fused via a multi-scale cross-attention mechanism, projecting them to generate high-dimension Bird's Eye View (BEV) feature maps $F^{bev} \in \mathbb{R}^{8\times8\times256}$.

We also introduce auxiliary learning tasks (BEV semantic segmentation and detection) to enrich the BEV representations with clear geometric and semantic boundaries. These geometrically constrained BEV features are then transformed

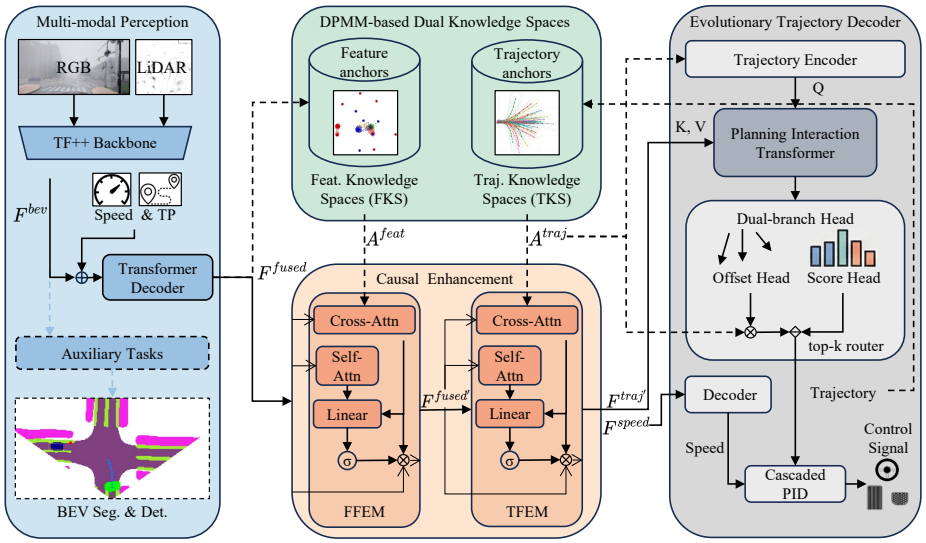

Fig. 2: Overview architecture of our proposed method.

and concatenated with the ego-vehicle's current velocity feature and target point feature. Finally, a transformer decoder equipped with 11 learnable queries pools these panoramic features into a highly compact fused scene representation vector $F^{fused} \in \mathbb{R}^{11\times256}$.

3.2 Dynamic Dual Knowledge Spaces

To overcome the memory bottleneck of fixed-capacity networks under continuous tasks, we employ the Dirichlet process mixture model (DPMM) [28] to construct explicit and implicit dual knowledge spaces in a dynamic manner. As a typical Bayesian non-parametric statistical model, DPMM allows the data to spontaneously dictate the number of cluster components, a property naturally suited to scenarios in lifelong learning where unknown data continuously flow in while the boundary is unknown.

DPMM employs the Dirichlet process as a prior to characterize how data are generated from an infinite number of potential clusters. The Dirichlet process can be formalized as $G \sim DP(\alpha, H)$, where G is a random probability measure over the parameter space, H is the base distribution that represents the prior expected parameter distribution, and α is the concentration parameter, which controls the aggressiveness of generating new clusters. Expressed using the stick-breaking process, the generative process of DPMM can be represented as follows:

$$\theta_k \mid \lambda \sim H(\lambda), \quad \pi \mid \alpha \sim \mathrm{GEM}(\alpha), \quad v_i \mid \pi \sim \mathrm{Cat}(\pi), \quad x_i \mid v_i \sim F\left(\theta_{v_i}\right), \quad (1)$$

where θ_k is a latent variable drawn independently from a Dirichlet process prior, with G as its base distribution. The mixing proportions π is sampled from a generalized Ewens distribution (GEM). Variable v_i assigns data point x_i to a cluster and takes on the value k with probability π_k, which is drawn from a categorical distribution (Cat). Each data point x_i is sampled from the belonged distribution $F(\theta_{v_i})$. Clustering emerges naturally in the DPMM because observations that share the same latent parameter θ_k, which is drawn from a discrete distribution, are automatically grouped together.

In our approach, we assume that the parameters of each component in the DPMM adhere to a shared Normal-Wishart base distribution. For computational efficiency, we also assume that each active component in the DPMM can be represented as a multi-variate Gaussian with only a diagonal covariance matrix.

Our framework instantiates DPMM knowledge spaces across two different levels:

Feature Knowledge Space (FKS) is an implicit feature space used to cluster the fused features F^{fused} extracted by the backbone network. The mission of FKS is to extract and preserve latent topological causal structures in the environment. As the learning sequence progresses, this space automatically extracts the center point of each cluster as feature knowledge anchors, denoted as $A^{feat} \in \mathbb{R}^{K_f \times 2816}$, where K_f is the total number of environmental feature patterns currently identified by the DPMM.

Trajectory Knowledge Space (TKS) is an explicit kinematic space that directly clusters the ground truth expert trajectories from historical training data using DPMM. It constructs a physical prior action library covering maneuvers like lane changing, and sharp turns. The corresponding cluster centers are extracted as trajectory knowledge anchors, denoted as $A^{traj} \in \mathbb{R}^{K_t \times 20}$, where K_t is the dynamically evolving number of trajectory prototypes.

Given the intractability of exact posterior inference in DPMM, we adopt memoVB [16] as an efficient approximation. memoVB decomposes global sufficient statistics into mini-batch summations, enabling online coordinate ascent updates. It leverages the nonparametric nature of DPMM to dynamically adjust cluster counts via birth and merge heuristics, helping the model escape local optima and continuously adapt to evolving data streams. During training, the DPMM and the neural network are updated in an alternating manner.

3.3 Causal Feature Enhancement Module

There are unobservable confounders in the latent space that may cause spurious correlations between perception and action. To mitigate the potential influence of unobservable confounders, we introduce the causal feature enhancement module inspired by front-door adjustment [38, 48].

The core idea of front-door adjustment is to construct a front-door path between input X and output Y via a mediator variable M ($X \to M \to Y$), and M intercepts directed paths from X to Y, while there are no unblocked back-door paths from M to Y. Then even in the presence of unobserved confounder U

$(X \leftarrow U \rightarrow Y)$, the interventional distribution $P(Y|do(X))$ can still be identified and calculated via the front-door adjustment formula:

$$P(Y = y|do(X = x)) = \sum_m P(m|x) \sum_{x'} P(y|x', m)P(x'). \tag{2}$$

In our design, the sets of knowledge anchors generated by DPMM (A^{feat} and A^{traj}) serve as the discrete state space for this observed mediator variable M. This causal intervention process is divided into two cascaded sub-modules, both adopting a unified dual-attention and gated fusion architecture.

Fused Feature Enhancement Module (FFEM) processes the raw multimodal fused features $F^{fused} \in \mathbb{R}^{11\times256}$. First, the feature knowledge anchors A^{feat} are mapped to a latent space via a projection network to obtain $\hat{F}^{fused} \in \mathbb{R}^{K_f\times256}$. Next, a Self-attention layer extracts the internal dependencies of the current input features, followed by a cross-attention layer that uses the input features as queries (Q) and the projected knowledge anchors as keys (K) and values (V). This operation essentially calculates the expectation term $\sum_m P(m|x)$ in the formula, searching for the most fitting historical causal template for the current cluttered scene:

$$F^{enhan} = \text{Attn}(Q = F^{input}, K = V = F^{knowledge}) + F^{input}. \tag{3}$$

Finally, a learnable gating network built with a sigmoid-activated Multi-Layer Perceptron (MLP) adaptively calculates the fusion weight w, smoothly combining the raw input with the causally enhanced features:

$$\begin{cases} F^{\text{output}} = w \odot F^{\text{enhan}} + (1 - w) \odot F^{\text{input}}, \\ w = \sigma\left[\text{MLP}\left(F^{\text{enhan}}\right) + \text{MLP}\left(\text{Attn}\ \left(Q = K = V = F^{\text{input}}\right)\right)\right]. \end{cases} \tag{4}$$

The FFEM ultimately outputs the deconfounded enhanced fused features $F^{fused'} \in \mathbb{R}^{11\times256}$.

Trajectory Feature Enhancement Module (TFEM) receives the subset of the FFEM output specifically responsible for trajectory prediction $F^{traj} = F^{fused'}_{1:10} \in \mathbb{R}^{10\times256}$. TFEM transforms the geometric coordinate anchors A^{traj} from the trajectory knowledge space into high-dimensional spatiotemporal features $\hat{F}^{traj} \in \mathbb{R}^{K_t\times10\times256}$ via positional encoding and temporal extension projection. Subsequently, TFEM reuses the exact same cross-attention and gated intervention mechanism, outputting trajectory features containing causal kinematic constraints $F^{traj'} \in \mathbb{R}^{10\times256}$. Meanwhile, the speed feature $F^{speed} = F^{fused'}_{11}$ is passed through directly to the downstream speed decoder.

3.4 Evolutionary Trajectory Decoder

To address the architectural rigidity of traditional fixed-channel decoders in lifelong learning, we propose an evolutionary trajectory decoder driven by the

dynamic trajectory knowledge space. Leveraging the transformer's permutation invariance and sequence flexibility, the decoder maps heterogeneous trajectory anchors A^{traj} into dynamic planning token via a temporal embedding network. As the DPMM expands the cluster count K_t with accumulated experience, this token pool grows naturally to achieve unbounded knowledge acquisition. These tokens then serve as queries in a planning interaction transformer, performing cross-attention against scene context features to evaluate the relevance of historical driving patterns in parallel. For trajectory generation, a dual-branch decoupled prediction head replaces traditional autoregressive methods with a parallel strategy, where a coarse-grained branch computes selection scores $Y^{logits} \in \mathbb{R}^{K_t}$ for each anchor, while a fine-grained branch predicts geometric offsets $Y^{offsets} \in \mathbb{R}^{K_t \times 20}$ to refine the predicted coordinates. The Y^{logits} is then transferred into a temperature-scaled probability distribution $\hat{Y}^{probs}$. Finally, the system generates a candidate set $\hat{Y}^{trajs}$ by applying offsets to the original anchors, with the final execution output $\hat{Y}^{traj}$ selected via a Top-K routing mechanism based on $\hat{Y}^{probs}$. The core mathematical expression for decoding is as follows:

$$\begin{cases} \hat{Y}^{trajs} = \mathrm{MLP}\left(\mathrm{Attn}\left(Q = E\left(A^{traj}\right), K = V = F^{traj}\right)\right) + A^{traj}, \\ \hat{Y}^{probs} = \varphi\left[\mathrm{MLP}\left(\mathrm{Attn}\left(Q = E\left(A^{traj}\right), K = V = F^{traj}\right)\right), \tau\right], \\ \hat{Y}^{traj} = \hat{Y}^{trajs}_{TopK\left(\hat{Y}^{probs}, k\right)}, \end{cases} \tag{5}$$

where $\varphi[\cdot]$ is Softmax function and τ is the temperature factor.

3.5 Loss Function

The overall loss function is given by:

$$\mathcal{L} = \mathcal{L}_{sem} + \mathcal{L}_{det} + \mathcal{L}_{traj} + \mathcal{L}_{speed}, \tag{6}$$

where the BEV semantic segmentation loss $\mathcal{L}_{sem}$ and the target speed loss $\mathcal{L}_{speed}$ are both cross-entropy losses. The BEV detection loss $\mathcal{L}_{det}$ follows the pattern of CenterNet [12], which consists of heatmap loss, offset loss and size loss. The trajectory loss $\mathcal{L}_{traj}$ consists of three parts, as given below:

$$\mathcal{L}_{traj} = \mathcal{L}_{prob} + \mathcal{L}_{best} + \mathcal{L}_{weighted}. \tag{7}$$

For anchor selection probability loss $\mathcal{L}_{prob}$, we use KL divergence to minimize the difference between the predicted probability distribution and the real probability distribution, given by:

$$\begin{cases} \mathcal{L}_{prob} = D_{KL}\left(Y^{probs} \| \hat{Y}^{probs}\right), \\ Y^{probs} = \varphi\left[-\left\|\hat{Y}^{trajs} - Y^{traj}\right\|_2, \tau\right], \end{cases} \tag{8}$$

where the ground-truth anchor selection probability distribution Y^{probs} is defined by applying the softmax function to the negated distances between all predicted trajectories and the ground-truth trajectory Y^{traj}. The best-trajectory

loss $\mathcal{L}_{best}$ and the weighted-trajectory loss $\mathcal{L}_{weighted}$ are both computed using the smooth L1 loss. The former measures the deviation between the ground truth and the closest predicted trajectory, while the latter computes the weighted sum of deviations from all anchor trajectories, with weights being the ground-truth selection probabilities. This is formulated as follows:

$$\begin{cases} \mathcal{L}_{best} = SmoothL1(\hat{Y}^{traj} - Y^{traj}), \\ \mathcal{L}_{weighted} = Y^{probs} \cdot SmoothL1(\hat{Y}^{trajs} - Y^{traj}). \end{cases} \tag{9}$$

4 Experiments

4.1 Dataset and Metrics

Inspired by existing lifelong learning methods [31,36,53], we construct a streaming data training pipeline for lifelong learning based on the Bench2Drive benchmark [20]. The dataset comprises approximately 512K frames covering five key driving competencies: Emergency Braking, Traffic Sign Recognition, Merging, Overtaking, and Giving Way. We leverage this inherent categorization to define five sequential learning tasks, simulating a lifelong learning scenario where tasks are encountered in order. The data volume for each competency decreases sequentially, with approximately 184K, 146K, 92K, 78K, and only 11K frames, respectively. To comprehensively evaluate the lifelong learning capability of models, we organize the tasks in the aforementioned order corresponding to decreasing data volume and increasing learning difficulty, thereby rigorously testing the model's resistance to forgetting and its knowledge transfer ability.

In addition to the well-established evaluation criteria in CARLA [11] and Bench2Drive [20], we define a suite of metrics for lifelong learning that follows common practices in the field [23, 32, 37]. Our framework comprises three categories: vertical metrics for temporal stability, horizontal metrics for knowledge transferability, and comprehensive metrics for overall task proficiency. While the categories are inspired by classic works [33,36], the exact calculations are adapted to align with the Bench2Drive benchmark's success criteria and the nature of our tasks. Each metric is detailed in the following sections.

Vertical Dimension (Time-Series Stability Metrics):

- **Forgetting Ratio (FR) ↓:** Measures the degree of performance decay on historically learned tasks after the model learns subsequent new tasks, relative to the performance immediately after learning them. The formula is as follows, where $SR_{i,j}$ represents the success rate on task j after learning the i-th task. Lower values indicate stronger resistance to forgetting:

$$FR = \frac{1}{N-1} \sum_{i=1}^{N-1} \left(SR_{i,i} - SR_{N,i} \right) / SR_{i,i}. \tag{10}$$

- **Process Forgetting Ratio (PFR) ↓:** Captures the severity of drastic performance oscillations during the learning process caused by interference

from other tasks, measuring the degree of deviation from the historical best state:

$$PFR = \frac{1}{N-1} \sum_{j=1}^{N-1} \left(\frac{1}{N-j} \sum_{i=j+1}^{N} \left(H_{i,j} - SR_{i,j} \right) / H_{i,j} \right), \tag{11}$$

where $H_{i,j} = \max \{SR_{i,j}, i = 1, 2, \ldots, j-1\}$ represents the model's historical best performance on task j after training i th task.

Horizontal Dimension (Knowledge Transferability Metrics):

- **Forward Transfer (FT) ↑:** Evaluates the zero-shot generalization or facilitating capability of the currently accumulated causal knowledge pool for subsequent unseen tasks:

$$FT = \frac{1}{N-1} \sum_{i=1}^{N-1} \left(\frac{1}{N-i} \sum_{j=i+1}^{N} SR_{i,j} \right). \tag{12}$$

- **Backward Transfer (BT) ↑:** Quantifies the system's average maintenance or even reverse-enhancement level across all past task sets after continuous learning:

$$BT = \frac{1}{N-1} \sum_{i=2}^{N} \left(\frac{1}{i-1} \sum_{j=1}^{i-1} SR_{i,j} \right). \tag{13}$$

Comprehensive Overall Performance Metrics: We also adopt the official metrics such as driving score (DS), success rate (SR) and multi-ability success rate following the CARLA [11] and Bench2Drive [20].

- **Average Driving Score ↑:** Comprehensively considers route completion and penalty deductions, reflecting the overall performance of the method.
- **Average Success Rate ↑:** The ratio of routes finished within time limits and without any traffic infractions.
- **Average Multi-Ability Success Rate ↑:** The average success metric derived after decoupling five different dimensions of abilities, reflecting the overall balance of the method.

4.2 Implementation Details

We adopt different training strategies under different settings. For full-data learning, a two-stage approach is adopted. In the first stage, we pre-train the backbone with only auxiliary task losses $\mathcal{L}_{sem}$ and $\mathcal{L}_{det}$. It is then followed by a second stage where the entire model is trained with all losses. Each stage is conducted for 30 epochs. We use a variable learning rate from 3e-4 to 3e-5. For full-data learning, our model is trained on 4 NVIDIA L40 GPUs with a total batch size

of 64 for about 64 hours. The GPU memory usage of our model's inference is approximately 1.69 GB, and the time consumption for a single forward pass is about 26.29 ms. For lifelong learning, the first task follows the aforementioned two-stage protocol, while for subsequent tasks, the backbone is frozen and training proceeds in a single stage for 30 epochs. We reset the intrinsic parameters of optimizers before initiating the training on every new incoming task.

4.3 Main Results

Table 1: Lifelong learning results on Bench2Drive [20]. For every two rows of data, the top row is the result of baseline model and the bottom row is the result of ours.

After Task	Overall(%) ↑		Ability(%) ↑					
	DS	SR	Merge	Overtake	EmgBrake	GiveWay	TSign	Multi-Ab Mean
1(EmgBrake)	75.89	53.18	62.50	8.89	40.00	50.00	70.00	46.28
	79.88	56.82	43.75	20.00	90.00	50.00	79.47	56.64
2(Tsign)	74.31	52.27	42.50	15.56	83.33	50.00	78.42	53.69
	77.24	55.91	52.50	15.56	80.00	50.00	82.11	56.03
3(Merge)	72.57	48.18	53.75	11.11	51.67	50.00	74.74	48.25
	75.26	53.18	55.00	17.76	68.33	50.00	77.89	53.80
4(Overtake)	69.12	39.09	37.50	46.67	21.67	50.00	57.37	42.64
	72.09	45.00	42.50	40.00	41.67	50.00	65.26	47.89
5(GiveWay)	60.89	30.00	35.00	8.89	26.67	50.00	55.79	35.27
	68.97	42.73	36.25	22.22	51.67	50.00	70.00	46.03
Lifelong Metrics	Vert. Metrics ↓		Hori. Metrics ↑		Overall Metrics ↑			
	FR	PFR	FT	BT	Avg DS	Avg SR	Avg Multi-Ab SR	
	44.50	40.25	41.11	52.83	70.55	44.54	45.23	
	33.97	**29.8**	**42.88**	**79.63**	**74.69**	**50.73**	**52.08**	

The lifelong learning evaluation on Bench2Drive demonstrates that our framework substantially outperforms the baseline TF++ [17, 57] across all sequential tasks, as shown in Table 1. After the final task, our method achieves an average driving score of 74.69%, while the average success rate improves to 50.73%. On data-scarce 'GiveWay', our model maintains 68.97% driving score and 42.73% success rate, versus baseline's decline to 60.89% and 30%. Our approach also reduces the process forgetting ratio from 40.25% to 29.8% and raises backward transfer from 52.83% to 79.63%, confirming that the dynamic knowledge spaces and causal intervention effectively mitigate catastrophic forgetting and enable positive knowledge transfer.

To the best of our knowledge, DeLL is the first work to perform lifelong learning in the field of closed-loop end-to-end autonomous driving (E2E-AD). Therefore, we adapt Experience Replay (ER) [4] and PackNet [35] to the baseline model for fair comparison, denoted as "Baseline + ER" and "Baseline + PackNet". As shown in Table 2, ER [4] requires extra space to cache the dataset due to its rehearsal-based nature, while PackNet [35] requires twice the training time because of the additional network pruning step. It demonstrates that our

Table 2: Comparison with lifelong learning methods adapted to E2E-AD on Bench2Drive [20] dataset. Bold means best.

Method	Training Cost ↓		Vert. Metrics ↓		Hori. Metrics ↑		Overall Metrics ↑		
	Data	Time	FR	PFR	FT	BT	DS	SR	Multi-Ability
Baseline [17, 57]	× 1	× 1	44.50	40.25	41.11	52.83	70.55	44.54	45.23
Baseline + ER [4]	× 1.35	× 1	41.52	31.11	39.26	79.17	72.80	47.57	49.86
Baseline + PackNet [35]	× 1	× 2	34.11	**25.75**	42.81	74.50	74.46	50.65	51.64
DeLL (Ours)	× **1**	× **1**	**33.97**	29.80	**42.88**	**79.63**	**74.69**	**50.73**	**52.08**

approach DeLL outperforms baselines on the vast majority of metrics, while notably requiring less training cost.

Table 3: Full-data learning results of E2E-AD methods on Bench2Drive [20]. Bold stands for best and underlined for second best.

Method	Overall(%)↑		Ability(%)↑					
	DS	SR	Merge	Overtake	EmgBrake	GiveWay	TSign	Multi-Ab Mean
AD-MLP [55]	18.05	0.00	0.00	0.00	0.00	0.00	4.35	0.87
TCP [50]	40.70	15.00	16.18	20.00	20.00	10.00	6.69	14.63
VAD [22]	42.35	15.00	8.11	24.44	18.64	20.00	19.15	18.07
UniAD [15]	45.81	16.36	14.10	17.78	21.67	10.00	14.21	15.55
ThinkTwice [19]	62.44	31.23	27.38	18.42	35.82	50.00	54.23	37.17
DriveTransformer [21]	63.46	35.01	17.57	35.00	48.36	40.00	52.10	38.60
DriveAdapter [18]	64.22	33.08	28.82	26.38	48.76	50.00	56.43	42.08
DiffAD [49]	67.92	38.64	30.00	35.55	46.66	40.00	46.32	38.79
DriveMoE [52]	74.22	48.64	34.67	40.00	65.45	40.00	59.44	47.91
ORION [13]	77.74	54.62	25.00	71.11	78.33	30.00	69.00	54.72
TF++ [17, 57]	84.21	67.27	58.75	57.77	**83.33**	40.00	82.11	64.39
SimLingo [40]	85.07	67.27	54.01	57.04	**88.33**	53.33	**82.45**	67.03
HiP-AD [46]	86.77	**69.09**	50.00	**84.44**	**83.33**	40.00	72.10	65.98
DeLL (Ours)	**86.86**	68.63	**61.25**	62.22	80.00	**60.00**	81.05	**68.90**

Under the full-data training paradigm, our method achieves state-of-the-art performance among all compared end-to-end models, as shown in Table 3. It attains the highest driving score of 86.86% and also leads in the average multi-ability success rate with 68.9%, reflecting well-rounded competence across diverse driving scenarios. These results confirm that the architectural innovations introduced for lifelong learning also enhance representational power and causal reasoning in static training settings.

4.4 Ablation Study

Ablation experiments quantify the contribution of each core component within our framework, as shown in Table 4. The evolutionary trajectory decoder proves indispensable for adaptive planning. Its removal reduces the average driving score to 72.94% and backward transfer to 72.21%. This suggests that the dynamically expanding trajectory anchor pool continually improves driving performance and

Table 4: Ablation study on Bench2Drive [20] under lifelong learning setting

Method	Overall Metrics(%) ↑			Vert. Metrics(%) ↓		Hori. Metrics(%) ↑	
	Avg DS	Avg SR	Avg Multi-Ab SR	FR	PFR	BT	FT
Baseline Model	70.55	44.54	45.23	44.50	40.25	52.83	41.11
w/o ET Dec.	72.94	49.82	50.74	33.12	31.76	72.21	42.98
w/o TFEM	73.00	48.36	49.92	36.43	32.86	77.14	40.04
w/o FFEM	73.10	49.33	49.02	38.33	30.49	77.32	36.59
Full Model	74.69	50.73	52.08	33.97	29.80	79.63	42.88

*w/o: full model without a certain module. TF++ [17, 57] is used as the baseline model.

is crucial for effective knowledge preserving. The TFEM plays a vital role in imposing causally consistent kinematic constraints. Without it, the average driving score falls to 73%, while the process forgetting ratio rises to 32.86%. This degradation confirms that enhancing trajectory features with front-door adjustment helps preserve maneuver-specific knowledge. The FFEM is critical for causal feature purification, as its removal causes the forgetting ratio to increase to 38.33% and forward transfer to drop to 36.59%, indicating that deconfounding raw multimodal features is essential for both knowledge retention and generalization to new tasks. The full model achieves the best performance, demonstrating that the synergistic combination of knowledge spaces and cascaded front-door adjustment is essential for robust lifelong autonomous driving.

4.5 Visualization

As shown in Fig. 3, TF++ [17, 57] initially learns to decelerate and follow a slow-moving bicycle ahead after training on Task 1 (1a), but subsequent training causes it to forget this speed control strategy, leading to a collision (1b, 1c). Our method also learns timely deceleration and low-speed following initially (2a) but does not change lanes to overtake due to the absence of training data (2b). It acquires the lane-changing overtaking skill after training on Task 4 (2c). TF++ and our method are both unable to perform parking spot exit maneuvers during the early stages of training (1d, 2d). After learning from Task 3, both methods acquire this capability (1e, 2e). However, following subsequent training, TF++ exhibits incorrect velocity predictions, potentially having learned a mistaken causal relationship between 0 velocity and a stationary vehicle ahead (1f). In contrast, our method continues to predict velocity correctly under the same conditions (2f). TF++ cannot reliably master lane merging (3a, 3b, 3c), whereas our method is able to decisively merge by exploiting traffic gaps on (4a) and gradually learns to decelerate in dense traffic before accelerating to complete lane changes (4b, 4c), illustrating the progressive accumulation of driving knowledge. When facing unseen scenarios at the early stage of learning, TF++'s planned trajectories often conflict with nearby vehicles (3d, 3e), while our method generates reasonable trajectories (4d, 4e), indicating forward transfer of knowledge. After learning subsequent tasks, TF++ forgets to decelerate at stop signs (3f), while our method retains this ability (4f), demonstrating resistance to forgetting.

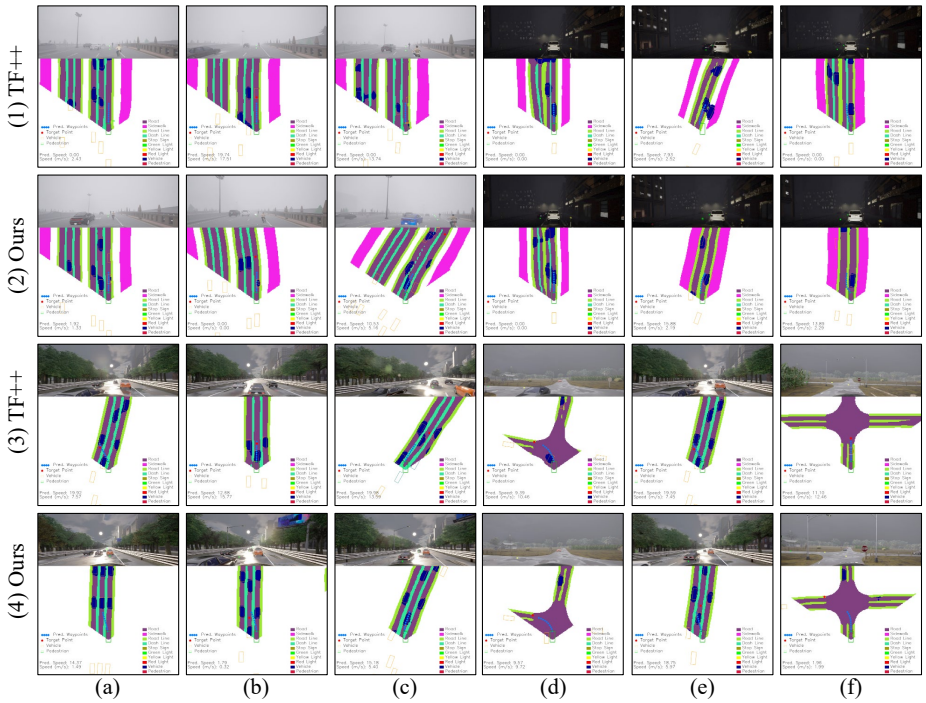

Fig. 3: Lifelong performance on CARLA benchmark DEV10 [21].

Fig. 4 presents the clustering results of dynamic knowledge spaces during the learning process, including numbers, IDs and visualizations. In particular, we sample 50 data points per cluster from the feature knowledge space and project them into 2D using t-SNE [27] for visualization, with colors indicating their associated driving ability. Notably, some driving capabilities encompass multiple clusters. While most clusters are clearly separated, a small number of clusters from different capabilities exhibit spatial proximity or partial overlap, indicating coupling relationships among distinct driving abilities.

5 Conclusion

In this paper, we propose DeLL, a deconfounded lifelong learning framework for end-to-end autonomous driving. Our method introduces dual dynamic knowledge spaces based on DPMM to incrementally preserve and update both latent feature representations and explicit trajectory priors. By leveraging these knowledge anchors as mediators in a causal front-door adjustment mechanism, the framework effectively mitigates spurious correlations caused by unobserved confounders. Additionally, an evolutionary trajectory decoder enables non-autoregressive, parallel planning. Extensive experiments in CARLA demonstrate that DeLL achieves

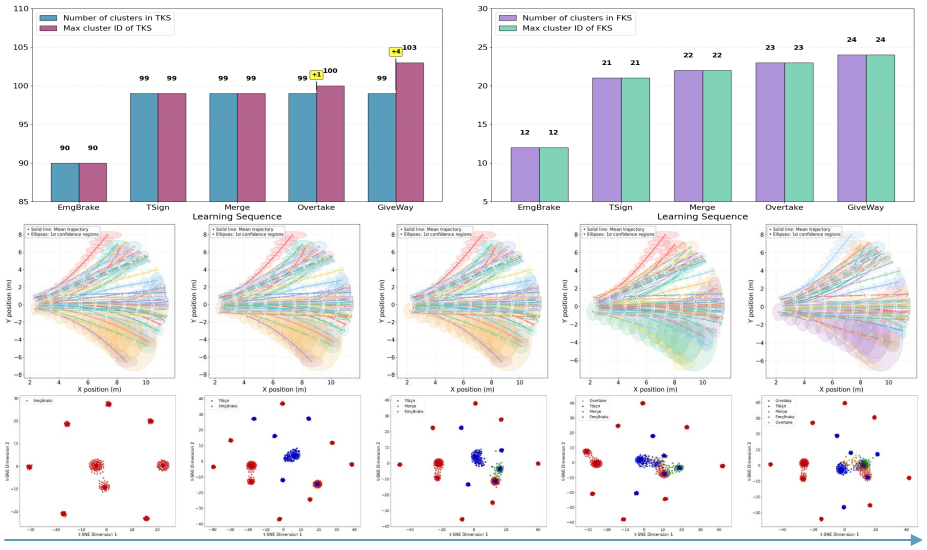

Fig. 4: Clustering results of dynamic knowledge spaces during the learning process.

superior driving performance, significantly reduces catastrophic forgetting, and enhances knowledge transfer across sequential tasks.

Despite its effectiveness, our method has certain limitations. The alternating optimization between DPMM updates and network training introduces computational overhead and training cost. In addition, the current framework operates within simulated environments, so the domain gap between CARLA and real-world deployment remains an open challenge.

Acknowledgements

This paper is supported by the National Natural Science Foundation of China under Grants No. 62473295, No. 62233013 and No. 62333017.

References

1. Aljundi, R., Babiloni, F., Elhoseiny, M., Rohrbach, M., Tuytelaars, T.: Memory aware synapses: Learning what (not) to forget. In: ECCV. pp. 139–154 (2018)
2. Aljundi, R., Chakravarty, P., Tuytelaars, T.: Expert gate: Lifelong learning with a network of experts. In: CVPR. pp. 3366–3375 (2017)
3. Castro, F.M., Marín-Jiménez, M.J., Guil, N., Schmid, C., Alahari, K.: End-to-end incremental learning. In: ECCV. pp. 233–248 (2018)
4. Chaudhry, A., Rohrbach, M., Elhoseiny, M., Ajanthan, T., Dokania, P.K., Torr, P.H., Ranzato, M.: On tiny episodic memories in continual learning. arXiv preprint arXiv:1902.10486 (2019)

5. Chen, D., Krähenbühl, P.: Learning from all vehicles. In: CVPR. pp. 17222–17231 (2022)
6. Chen, D., Zhou, B., Koltun, V., Krähenbühl, P.: Learning by cheating. In: CoRL. pp. 66–75. PMLR (2020)
7. Chen, L., Wu, P., Chitta, K., Jaeger, B., Geiger, A., Li, H.: End-to-end autonomous driving: Challenges and frontiers. TPAMI **46**(12), 10164–10183 (2024)
8. Chitta, K., Prakash, A., Geiger, A.: Neat: Neural attention fields for end-to-end autonomous driving. In: ICCV. pp. 15793–15803 (Oct 2021)
9. Chitta, K., Prakash, A., Jaeger, B., Yu, Z., Renz, K., Geiger, A.: Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. TPAMI **45**(11), 12878–12895 (2022)
10. De Haan, P., Jayaraman, D., Levine, S.: Causal confusion in imitation learning. In: NeurIPS. vol. 32 (2019)
11. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: CARLA: An open urban driving simulator. In: CoRL. pp. 1–16. PMLR (2017)
12. Duan, K., Bai, S., Xie, L., Qi, H., Huang, Q., Tian, Q.: Centernet: Keypoint triplets for object detection. In: ICCV. pp. 6569–6578 (2019)
13. Fu, H., Zhang, D., Zhao, Z., Cui, J., Liang, D., Zhang, C., Zhang, D., Xie, H., Wang, B., Bai, X.: Orion: A holistic end-to-end autonomous driving framework by vision-language instructed action generation. In: ICCV. pp. 24823–24834 (2025)
14. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Communications of the ACM **63**(11), 139–144 (2020)
15. Hu, Y., Yang, J., Chen, L., Li, K., Sima, C., Zhu, X., Chai, S., Du, S., Lin, T., Wang, W., et al.: Planning-oriented autonomous driving. In: CVPR. pp. 17853–17862 (2023)
16. Hughes, M.C., Sudderth, E.: Memoized online variational inference for dirichlet process mixture models. In: NeurIPS. vol. 26 (2013)
17. Jaeger, B., Chitta, K., Geiger, A.: Hidden biases of end-to-end driving models. In: ICCV. pp. 8240–8249 (2023)
18. Jia, X., Gao, Y., Chen, L., Yan, J., Liu, P.L., Li, H.: Driveadapter: Breaking the coupling barrier of perception and planning in end-to-end autonomous driving. In: ICCV. pp. 7953–7963 (2023)
19. Jia, X., Wu, P., Chen, L., Xie, J., He, C., Yan, J., Li, H.: Think twice before driving: Towards scalable decoders for end-to-end autonomous driving. In: CVPR. pp. 21983–21994 (2023)
20. Jia, X., Yang, Z., Li, Q., Zhang, Z., Yan, J.: Bench2Drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. In: NeurIPS (2024)
21. Jia, X., You, J., Zhang, Z., Yan, J.: Drivetransformer: Unified transformer for scalable end-to-end autonomous driving. In: ICLR. pp. 67227–67243 (2025)
22. Jiang, B., Chen, S., Xu, Q., Liao, B., Chen, J., Zhou, H., Zhang, Q., Liu, W., Huang, C., Wang, X.: Vad: Vectorized scene representation for efficient autonomous driving. In: ICCV. pp. 8340–8350 (2023)
23. Jiang, M., Fan, J., Li, F.: Advances in continual learning: A comprehensive review. Expert Systems with Applications **294**, 128739 (2025)
24. Kemker, R., Kanan, C.: Fearnet: Brain-inspired model for incremental learning. In: ICLR (2018)
25. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)

26. Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al.: Overcoming catastrophic forgetting in neural networks. Proceedings of the national academy of sciences **114**(13), 3521–3526 (2017)
27. Laurens, V.D.M., Hinton, G.: Visualizing data using t-sne. Journal of Machine Learning Research **9**(2605), 2579–2605 (2008)
28. Li, Y., Schofield, E., Gönen, M.: A tutorial on dirichlet process mixture modeling. Journal of Mathematical Psychology **91**, 128–144 (2019)
29. Li, Z., Hoiem, D.: Learning without forgetting. TPAMI **40**(12), 2935–2947 (2017)
30. Liao, B., Chen, S., Yin, H., Jiang, B., Wang, C., Yan, S., Zhang, X., Li, X., Zhang, Y., Zhang, Q., et al.: Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In: CVPR. pp. 12037–12047 (2025)
31. Lin, Y., Li, Z., Du, G., Zhao, X., Gong, C., Wang, X., Lu, C., Gong, J.: H2c: Hippocampal circuit-inspired continual learning for lifelong trajectory prediction in autonomous driving. TITS pp. 1–18 (2026)
32. Liu, B., Zhu, Y., Gao, C., Feng, Y., Liu, Q., Zhu, Y., Stone, P.: Libero: Benchmarking knowledge transfer for lifelong robot learning. In: NeurIPS. vol. 36, pp. 44776–44791 (2023)
33. Lopez-Paz, D., Ranzato, M.: Gradient episodic memory for continual learning. In: NeurIPS. vol. 30 (2017)
34. Luo, Y., Yin, L., Bai, W., Mao, K.: An appraisal of incremental learning methods. Entropy **22**(11), 1190 (2020)
35. Mallya, A., Lazebnik, S.: Packnet: Adding multiple tasks to a single network by iterative pruning. In: CVPR. pp. 7765–7773 (2018)
36. Meng, Y., Bing, Z., Yao, X., Chen, K., Huang, K., Gao, Y., Sun, F., Knoll, A.: Preserving and combining knowledge in robotic lifelong reinforcement learning. Nature Machine Intelligence pp. 1–14 (2025)
37. Parisi, G.I., Kemker, R., Part, J.L., Kanan, C., Wermter, S.: Continual lifelong learning with neural networks: A review. Neural Networks **113**, 54–71 (2019)
38. Pearl, J., Glymour, M., Jewell, N.P.: Causal inference in statistics: A primer. John Wiley & Sons (2016)
39. Radosavovic, I., Kosaraju, R.P., Girshick, R., He, K., Dollár, P.: Designing network design spaces. In: CVPR. pp. 10428–10436 (2020)
40. Renz, K., Chen, L., Arani, E., Sinavski, O.: Simlingo: Vision-only closed-loop autonomous driving with language-action alignment. In: CVPR (2025)
41. Renz, K., Chitta, K., Mercea, O.B., Koepke, A.S., Akata, Z., Geiger, A.: Plant: Explainable planning transformers via object-level representations. In: CoRL (2022)
42. Ruan, K.: When Causality Meets Autonomy: Causal Imitation Learning to Unravel Unobserved Influences in Autonomous Driving Decision-Making. Columbia University (2024)
43. Rusu, A.A., Rabinowitz, N.C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., Hadsell, R.: Progressive neural networks. arXiv preprint arXiv:1606.04671 (2016)
44. Shao, H., Wang, L., Chen, R., Li, H., Liu, Y.: Safety-enhanced autonomous driving using interpretable sensor fusion transformer. In: CoRL. pp. 726–737. PMLR (2023)
45. Shin, H., Lee, J.K., Kim, J., Kim, J.: Continual learning with deep generative replay. In: NeurIPS. vol. 30 (2017)
46. Tang, Y., Xu, Z., Meng, Z., Cheng, E.: Hip-ad: Hierarchical and multi-granularity planning with deformable attention for autonomous driving in a single decoder. In: ICCV. pp. 25605–25615 (2025)

47. Van de Ven, G.M., Tuytelaars, T., Tolias, A.S.: Three types of incremental learning. Nature Machine Intelligence **4**(12), 1185–1197 (2022)
48. Wang, L., He, Z., Dang, R., Shen, M., Liu, C., Chen, Q.: Vision-and-language navigation via causal learning. In: CVPR. pp. 13139–13150 (2024)
49. Wang, T., Zhang, C., Qu, X., Li, K., Liu, W., Huang, C.: Diffad: A unified diffusion modeling approach for autonomous driving. arXiv preprint arXiv:2503.12170 (2025)
50. Wu, P., Jia, X., Chen, L., Yan, J., Li, H., Qiao, Y.: Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. In: NeurIPS. vol. 35, pp. 6119–6132 (2022)
51. Yang, Y., Chen, D., Qin, T., Mu, X., Xu, C., Yang, M.: E2e parking: Autonomous parking by the end-to-end neural network on the carla simulator. In: 2024 IEEE Intelligent Vehicles Symposium (IV). pp. 2375–2382. IEEE (2024)
52. Yang, Z., Chai, Y., Jia, X., Li, Q., Shao, Y., Zhu, X., Su, H., Yan, J.: Drivemoe: Mixture-of-experts for vision-language-action model in end-to-end autonomous driving. In: CVPR. pp. 10678–10688 (2026)
53. Yao, H., Li, P., Jin, B., Zheng, Y., Liu, A., Mu, L., Su, Q., Zhang, Q., Chen, Y., Li, P.: Lilodriver: A lifelong learning framework for closed-loop motion planning in long-tail autonomous driving scenarios. arXiv preprint arXiv:2505.17209 (2025)
54. Yao, X., Huang, T., Wu, C., Zhang, R.X., Sun, L.: Adversarial feature alignment: Avoid catastrophic forgetting in incremental task lifelong learning. Neural computation **31**(11), 2266–2291 (2019)
55. Zhai, J.T., Feng, Z., Du, J., Mao, Y., Liu, J.J., Tan, Z., Zhang, Y., Ye, X., Wang, J.: Rethinking the open-loop evaluation of end-to-end autonomous driving in nuscenes. arXiv preprint arXiv:2305.10430 (2023)
56. Zhai, M., Chen, L., Tung, F., He, J., Nawhal, M., Mori, G.: Lifelong gan: Continual learning for conditional image generation. In: ICCV. pp. 2759–2768 (2019)
57. Zimmerlin, J., Beißwenger, J., Jaeger, B., Geiger, A., Chitta, K.: Hidden biases of end-to-end driving datasets. arXiv preprint arXiv:2412.09602 (2024)